UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 09/620,771
Filing Date: July 21, 2000
Appellant(s): MEREDITH ET AL.

Gregory Lucius Meredith et al.
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 11/02/2007 appealing from the Office action mailed 11/29/2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal, beyond that identified by the Appellant.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Hsu et al. (U.S. 5,581,691)

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-22, 26, 28-40, 42-44, 46, and 48-52 are rejected under 35 U.S.C. 102(b) as being anticipated by Hsu et al. (U.S. 5,581,691).

As per **claim 11**, Hsu et al. discloses a method of executing a schedule, the schedule comprising a schedule state, at least one transaction having a workflow action associated therewith, the action having a latency attribute associated therewith, the method comprising:

initiating the workflow action according to the schedule (See column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-62, column 12, lines 1-22, wherein workflow events are managed by the system and the actions of each event are instantiated when input event signals are received);

comparing the latency attribute with a latency threshold (See column 3, lines 10-25, column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 8, lines 30-45, column 10, lines 5-20, column 11, lines 50-65, wherein actions within the workflow are created or instantiated only when a sufficient amount of signals are received to start such an action. Thus an attribute of the action (an input condition) is compared to a threshold of received signals);

selectively storing the schedule state in a storage medium based on the latency comparison (See column 3, lines 10-25, column 5, lines 9-15, column 13, lines 25-40 and 50-60, column 15, lines 5-15, wherein state and log information is stored in the system).

As per **claim 12**, Hsu et al. teaches creating an association between the stored schedule state and a signal (See column 3, lines 10-25, column 5, lines 23-30 and 60-67, column 8, lines 30-45, column 9, lines 10-25, wherein input conditions are defined and stored that represent the required input signals needed to cause the next step to occur. See also column 13, lines 25-40 and 50-60).

As per **claim 13**, Hsu et al. teaches suspending execution of the schedule based on the latency comparison (See column 3, lines 10-25, which discloses waiting to resume execution while waiting for an input. See also column 7, lines 54-56, which discloses timing out an execution).

As per **claim 14**, Hsu et al. teaches selectively de-allocating resources associated with the schedule after storing the schedule state in the storage medium (See column 5, lines 45-60, column 7, lines 25-55, column 12, lines 1-25, column 13, lines 10-15 and 50-65, wherein resources are dynamically allocated and wherein when an activity stops that is using the resource, it is de-allocated and available to another activity).

As per **claim 15**, Hsu et al. teaches selectively resuming execution of the schedule based on the signal (See column 3, lines 10-25, where execution is resumed when the input is received. See also column 4, lines 60-67, column 5, lines 60-67, column 7, lines 5-20 and 54-56).

As per **claim 16**, Hsu et al. discloses selectively allocating computer system resources for execution of the schedule based on the signal (See column 5, lines 50-60, column 7, lines 25-40,

wherein resources are dynamically allocated) and selectively resuming execution of the schedule based on the signal (See column 3, lines 10-25, where execution is resumed. See also column 4, lines 60-67, column 5, lines 60-67, column 7, lines 5-20 and 54-56).

As per **claim 17**, Hsu et al. teaches wherein the schedule includes a plurality of actions and at least one of the actions has an associated latency attribute (See column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-65, column 12, wherein workflows include a plurality of actions. See also column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 10, lines 5-20, wherein input conditions (attributes) are associated with the actions that make up the workflow).

As per **claim 18**, Hsu et al. teaches wherein the latency attribute represents an estimated latency for completion of the associated action (See column 7, lines 8-24 and 54-56, and column 13, lines 23-40, wherein timeouts occur when an action surpasses an expected duration).

As per **claim 19**, Hsu et al. teaches adjusting at least one of the latency attributes according to a variable (See column 13, line 30-column 14, line 25, which discloses changing n attribute to reflect that it is completed).

As per **claim 20**, Hsu et al. teaches wherein the variable is related to an actual latency for completion of the associated workflow action (See column 13, line 30-column 14, line 25, wherein the variable of completion is related to the fact that the action has actually completed).

As per **claim 21**, Hsu et al. teaches wherein the latency attributes have a class associated therewith, and wherein the class indicates a group of actions (See column 3, lines 10-25, column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 8, lines 30-45, column 10, lines

5-20, column 11, lines 50-65, which discloses sets of latency attributes, like input and output conditions, time setting conditions, etc.).

As per **claim 22**, Hsu et al. teaches providing a plurality of latency thresholds, wherein each latency threshold has a class associated therewith, and selectively comparing a latency attribute with a latency threshold having the same class upon initiating the action associated with the latency attribute (See column 3, lines 10-25, column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 8, lines 30-45, column 10, lines 5-20, column 11, lines 50-65, wherein actions within the workflow are created or instantiated only when a sufficient amount of signals are received to start such an action. Thus an attribute of the action (an input condition) is compared to a threshold of received signals).

As per **claim 26**, Hsu et al. discloses selectively storing the schedule state in a database schema based on the latency comparison (See column 6, lines 20-37, which discloses a database schema. See column 3, lines 10-25, column 5, lines 9-15, column 13, lines 25-40 and 50-60, column 15, lines 5-15, wherein state and log information is stored in the system).

As per **claim 28**, Hsu et al. teaches wherein the workflow action has a compensation parameter associated therewith, further comprising selectively compensating the workflow action based on the compensation parameter, a transaction boundary within the schedule, and a state associated with another action within the schedule (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein a workflow action is compensated).

As per **claim 29**, Hsu et al. teaches selectively compensating a first workflow action according to a transaction boundary within the schedule and a compensation parameter associated with the first workflow action, based on abortion of a second action within the

schedule (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein when a second action is aborted, the compensation routines are enacted).

Claims 1-6, 7-8, 9, and 10 recite equivalent limitations to claims 11-16, 19-20, 11, and 25, respectively, and are therefore rejected using the same art and rationale set forth above.

As per **claim 30**, Hsu et al. teaches a method of executing a schedule, the schedule comprising a schedule state, at least one transaction with a workflow action associated therewith, the method comprising:

initializing a workflow action within the schedule (See column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-62, column 12, wherein workflow events are managed by the system and the actions of each event are instantiated when input event signals are received);

comparing a latency attribute associated with the workflow action and a latency threshold (See column 3, lines 10-25, column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 8, lines 30-45, column 10, lines 5-20, column 11, lines 50-65, wherein actions within the workflow are created or instantiated only when a sufficient amount of signals are received to start such an action. Thus an attribute of the action (an input condition) is compared to a threshold of received signals);

executing the action if the latency attribute does not exceed the latency threshold (See column 3, lines 10-25, column 5, lines 23-30, 35-50, and 60-67, column 7, lines 50-60, column 8, lines 30-45, column 10, lines 5-20, column 11, lines 50-65, wherein the action is executed as long as it does not exceed a threshold specifying a time to timeout);

dehydrating the schedule if the latency attribute exceeds the latency threshold (See column 3, lines 10-25, which discloses waiting to resume execution while waiting for an input. See also column 7, lines 54-56, which discloses timing out an execution).

As per claim 31, Hsu et al. teaches wherein dehydrating the schedule further comprises storing the schedule state to a storage medium, creating a proxy between the stored schedule state and a message, suspending execution of the schedule pending the expected workflow action, and restoring the schedule and resuming execution of the schedule based on receipt of the message (See column 3, lines 10-25, which discloses waiting to resume execution while waiting for an input. See also column 15, line 50-column 16, line 2 and lines 30-50. See also column 7, lines 54-56, which discloses timing out an execution).

Claims 32-37 are computer-readable medium versions of the method of claims 1-6, respectively. Since the disclosure of Hsu et al. is embodied on a computer-readable medium, claims 32-37 are rejected using the same art and rationale as relied upon in the rejection of claims 1-6, respectively.

As per **claims 38 and 39**, claims 38 and 39 are computer-readable medium versions of the method of claims 28 and 29, respectively. Since the disclosure of Hsu et al. is embodied on a computer-readable medium, claims 38 and 39 are rejected using the same art and rationale as relied upon in the rejection of claims 28 and 29, respectively.

As per **claim 40**, Hsu et al. discloses a method of executing a transaction having an associated transaction boundary and a workflow action, wherein the workflow action has an action state and a compensation parameter associated therewith, the method comprising:

recognizing a transaction boundary associated with the transaction (See abstract, column 1, lines 15-31, column 14, lines 1-18, which discloses transaction processing. See also See column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-62); and

selectively compensating at least a first workflow action according to the transaction boundary and the compensation parameter based on abortion of a second action (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein a workflow action is compensated).

As per **claim 42**, Hsu et al. teaches selectively compensating at least a first action according to the transaction boundary and the compensation parameter upon abortion of a second action, and further according to the action state associated with the first action (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein when a second action is aborted, the compensation routines are enacted, for the aborted action as well as previously enacted actions).

As per **claim 43**, Hsu et al. discloses selectively compensating at least a first action according to the transaction boundary and the compensation parameter upon abortion of a second action, if the first action has committed (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein when a second action is aborted, the compensation routines are enacted for the aborted action as well as previously enacted actions).

As per **claim 44**, Hsu et al. teaches wherein the compensation step further comprises instantiating at least one object (See column 7, lines 8-25).

As per **claim 46**, Hsu et al. discloses a computer-readable medium having computer-executable instructions for:

executing a schedule, the schedule comprising a schedule state, at least one workflow action, and at least one transaction with an associated transaction boundary, the workflow action including an action state and a compensation parameter associated therewith (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, which discloses transactions and states with compensation parameters. See also column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-62, column 12, lines 1-22, wherein workflow events are managed by the system and the actions of each event are instantiated when input event signals are received);

recognizing the transaction boundary within the schedule (See abstract, column 1, lines 15-31, column 14, lines 1-18, which discloses transaction processing. See also See column 3, lines 1-25, column 6, lines 20-35, column 8, lines 30-45, column 11, lines 33-62); and

selectively compensating at least a first workflow action within the schedule according to a transaction boundary within the schedule, and a compensation parameter associated with the first action based on abortion of a second workflow action (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein a workflow action is compensated, the compensation routines are enacted for the aborted action as well as previously enacted actions).

Claims 48-49 are computer-readable medium versions of the method of claims 42-43, respectively. Since the disclosure of Hsu et al. is embodied on a computer-readable medium, claims 48-49 are rejected using the same art and rationale as relied upon in the rejection of claims 42-43, respectively.

Claim 50 recites substantially similar limitations to claim 1 and is therefore rejected using the same art and rationale relied upon above.

As per **claim 51**, Hsu et al. teaches a schedule having a schedule state, a workflow action with an associated action state, and at least one inter-business transaction with a transaction boundary, a compensation parameter, a compensation routine, and a transaction state associated therewith, a method of selectively compensating the transaction during the execution of a schedule comprising:

determining the action state of the workflow action (See column 3, lines 10-25, column 5, lines 23-30 and 60-67, column 8, lines 30-45, column 9, lines 10-25, column 13, lines 25-40 and 50-60);

if the action state is aborted, determining the relationship of the workflow action and the transaction based on a transaction boundary (See abstract, column 1, line 15-30, column 7, lines 5-25 and 50-60, column 14, lines 1-20);

if the action state is aborted, and if the workflow action and transaction are related according to the transaction boundary, determining the transaction state of the transaction (See abstract, column 1, line 15-30, column 7, lines 5-25 and 50-60, column 14, lines 1-20);

if the action state is aborted and if the workflow action and the transaction are related according to the transaction boundary, and if the transaction state is committed, performing an operation according to the compensation routine associated with the transaction (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein when a second action is aborted, the compensation routines are enacted for the aborted action as well as previously enacted actions).

As per **claim 52,** Hsu et al. teaches a schedule having a schedule state, first and second transactions with associated transaction boundaries, transactions stated, compensation

parameters, and compensation routines, and first and second workflow actions with associated action states, compensation parameters, and compensation routines, a method of selectively compensating a first workflow action or transaction during the execution of a schedule comprising:

determining the state of one of the second workflow action and the second transaction (See column 3, lines 10-25, column 5, lines 23-30 and 60-67, column 8, lines 30-45, column 9, lines 10-25, column 13, lines 25-40 and 50-60);

if the state of one of the second workflow action and the second transaction is aborted, determining the relationship of the first workflow action and the transaction with the second workflow action and transaction based on a transaction boundary (See abstract, column 1, line 15-30, column 7, lines 5-25 and 50-60, column 14, lines 1-20);

if the state of one of the second workflow action and the second transaction is aborted, and one of the first workflow action and transaction are related to one of the second workflow action and transaction according to the transaction boundary, determining the state of one of the first workflow action and transaction (See abstract, column 1, line 15-30, column 7, lines 5-25 and 50-60, column 13, lines 25-40 and 50-60, column 14, lines 1-20); and

if the state of one of the second workflow action and the second transaction is aborted and if one of the first workflow action and transaction are related to one of the second workflow action and the transaction according to the transaction boundary, and if the state of one of the first workflow action and transaction is committed, performing an operation according to the compensation routine associated with one of the first workflow action and transaction (See column 7, lines 8-25, column 13, lines 25-40 and 50-60, column 14, lines 1-20, wherein when a

second action is aborted, the compensation routines are enacted for the aborted action as well as previously enacted actions).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 23-25, 27, and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hsu et al. (U.S. 5,581,691).

As per **claim 23**, Hsu et al. teaches editing tools to define workflows (See column 4, lines 54-60) and adjusting start conditions for an action in the workflow when resources are not available (See column 5, lines 50-60, column 7, lines 25-40). However, Hsu et al. does not expressly disclose adjusting at least one of the latency thresholds based on a variable.

Hsu et al. discloses a workflow management system including editing tools and the ability to create actions in a workflow. Hsu et al. further discloses a latency threshold, which identifies the input conditions that need to be satisfied. It is old and well known to allow a user to update and edit a stored plan (such as a workflow) after it has begun to execute in order to compensate for changes that have occurred or unpredicted circumstances. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to allow a user to adjust the latency thresholds in order to more efficiently allow the workflow to avoid timeouts, for example, by allowing the user to change the parameters of the system.

As per **claim 24**, Hsu et al. discloses wherein the variable is related to system resource utilization (See column 5, lines 50-60, column 7, lines 25-40, wherein resources are dynamically allocated, which discusses available and ready system resources).

Claim 25 recites substantially similar limitations to claim 23 and is therefore rejected using the same art and rationale set forth above.

As per **claim 27**, Hsu et al. teaches wherein the schedule state comprises a data structure and active data (See column 13, lines 25-50). Hsu et al. further discloses a database schema (See column 6, lines 20-35). However, Hsu et al. does not expressly disclose a schedule location.

Hsu et al. discloses a data structure and a database schema for the workflow management system. It is old and well known that databases have defined locations in which they store specific data. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include schedule location information concerning where the information about the workflow is stored in the database of the system of Hsu et al. in order to more effiently store and retrieve data of the system.

As per **claim 45**, Hsu et al. does not expressly disclose that the compensation step further comprises sending a message. Examiner takes official notice that is it is old and well known in workflow applications to alert a user as to occurrences of the system. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to send a message to the user of Hsu et al. in order to increase the usability of the system by alerting the user of actions being performed in the system.

(10) Response to Argument

In the Appeal Brief, Appellant makes the following arguments:

1) Hsu et al. does not teach or suggest comparing a latency attribute with a latency threshold and selectively storing data associated with a schedule in a storage medium based on the latency comparison;

2) the maximum amount of time allotted for executing a step in a workflow cannot be compared with comparing a latency attribute with a latency threshold

3) Hsu et al. does not teach or suggest selectively compensating at least a first workflow action according to the transaction boundary and a compensation parameter based on abortion of a second workflow action, as per claim 40;

4) Hsu et al. does not teach or suggest that if the action state is aborted, and if the workflow action and transaction are related according the transaction boundary, determining the transaction state of the transaction, let alone performing an operation according to the compensation routine associated with the transaction upon committing the transaction state, as per claims 51-52.

In response to argument 1), Examiner respectfully disagrees. Claim 1 recites:

a method of processing a workflow action within a schedule and having a latency attribute associated therewith, comprising:
initiating the workflow action;
comparing the latency attribute that is associated with the workflow action with a latency threshold;
selectively storing data associated with the schedule state in a storage medium based on the latency comparison.

Claim 11 recites substantially similar limitations:

a method of executing a schedule, the schedule comprising a schedule state, at least one transaction having a workflow action associated therewith, the action having a latency attribute associated therewith, the method comprising:
initiating the workflow action according to the schedule;
comparing the latency attribute with a latency threshold;

selectively storing the schedule state in a storage medium based on the latency comparison.

Hsu et al. does teach and suggest these claims. Hsu et al. teaches a system and method for executing and tracking the progress of long running work flows and recovering from system failures during the execution. The system is able to automatically recover from failure by storing sufficient data to determine the state just prior to failure and re-initiate processing of all interrupted work. See column 1, lines 55-65, and column 2, line 64-column 3, line 30. The workflow is represented in a flow description database as a set of steps with data flows between the steps. During execution of a workflow, each step is instantiated only when a defined set of input signals is received. When this happens, an instance is created and executed, resulting in one or more output signals. Log records are stored upon instantiation, execution, and termination of each step of the workflow and outputs are logged. Thus, the workflow state is stored for recovery and resumption upon failure with no loss of work. See also column 4, lines 59-67, which discusses workflow states and compensation and restarting of workflows.

Hsu et al. does teach and suggest claims 1 and 11. Hsu et al. teaches that a workflow action is initiated in at least column 3, lines 1-25, and column 8, lines 30-45, wherein workflow events are managed by the system and the actions of each event are instantiated when input event signals are received. Hsu et al. then teaches that actions within the workflow are created or instantiated only when a sufficient amount of signals are received to start such an action. Thus an attribute of the action (an input condition) is compared to a threshold of received signals. See column 3, lines 10-25, column 7, lines 50-60, column 10, lines 5-20, column 13, lines 30-51, and column 19, lines 25-36. Latency in computers is a time determination. Hsu teaches that each step/action in the workflow has an attribute (or time setting/time out limits) associated with it.

The workflow step/action times out when the time surpasses a latency threshold (or a point that must not be exceeded). See again column 7, lines 50-60, column 10, lines 5-20, column 13, lines 30-51, and column 19, lines 25-36, which discuss time settings that equal the time at which the step/action will timeout if execution of the step/action has not completed, monitoring timeout limits, and a timeout duration that indicated the maximum amount of time that should be allocated for execution of a step/action. Thus, "comparing a latency attribute that is associated with the workflow action with a latency threshold" is comparing the attribute of the action (time out setting of the step) with a latency threshold (maximum amount of time that should be allocated for execution of a step/action).

Hsu et al. also teaches "selectively storing data associate with a schedule in a storage medium based on the latency comparison". Hsu et al. teaches that the system monitors the time taken by the action. When the comparison indicates that the threshold has been surpassed, the system suspends the action/step and stores information to the database for later restart. See column 3, lines 10-25, column 5, lines 9-15, column 13, lines 25-40 and 50-60, column 15, lines 5-15, column 17, lines 35-50, which discloses storing state and log information in the system. Thus, Hsu et al. selectively stores data associated with an execution schedule in the storage medium/database log of the system based on the workflow action/step timing out. In instances where timeouts do not occur, the system still logs the system inputs and outputs.

In response to argument 2), Examiner respectfully disagrees. Appellant states on page 7 of the Appeal Brief that "A latency attribute is associated with a dormant period that becomes active ***or*** may represent the estimated time a corresponding action will take to complete. Based

on the latency comparison, a decision is made to store the data." Examiner notes the use of alternative language in this statement. Examiner agrees that a latency attribute may represent the estimated time a corresponding action will take to complete. This is what is taught by Hsu et al., as explained above. Hsu et al. clearly teaches that each step/action in the workflow has an attribute (or time setting/time out limits) associated with it, where each action has an expected amount of time it will take. When it surpasses this time limit, it times out. See again column 7, lines 50-60, column 10, lines 5-20, column 13, lines 30-51, and column 19, lines 25-36. Thus, since Appellant states that a latency attribute may represent the estimated time a corresponding action will take to complete, Hsu et al. "maximum amount of time allotted for executing a step" reads on such a latency interpretation.

In response to argument 3), Examiner respectfully disagrees. Claim 40 recites:

> A method of executing a transaction having an associated transaction boundary and a workflow action, wherein the workflow action has an action state and a compensation parameter associated therewith, the method comprising:
> recognizing a transaction boundary associated with the transaction
> selectively compensating at least a first workflow action according to the transaction boundary and the compensation parameter based on abortion of a second action.

Hsu et al. teaches long running transactions (or system implemented workflows) that have boundaries associated therewith. See abstract, column 1, lines 15-31, column 2, line 45-column 3, line 30, column 13, lines 25-55, and column 14, lines 1-18, which discloses long running transaction processing. Transactions are described as having the ability to fail and that such interrupted transactions can be restarted. Thus, transactions have boundaries at which they will timeout, or fail, if execution of a step is not yet complete. See specifically column 13, lines 25-55, and column 14, lines 1-18. Examiner notes that boundary, in its broadest reasonable

interpretation, means border, edge, or limit. Thus, the boundary is the limit at which the long running transaction will fail.

Hsu et al. further teaches “selectively compensating at least a first workflow action according to the transaction boundary and the compensation parameter based on abortion of a second action”. In the broadest reasonable interpretation, this limitation requires that a first workflow action/step is compensated according to the transaction boundary (i.e. the limit being crossed and the transaction failing/timing out) and a compensation parameter (a factor or consideration that allows the system to adjust and recoup from the failure) based on the abortion (termination) of a second workflow action/step. See column 2, line 64-column 3, line 7 and lines 25-30, column 4, lines 65-67, column 7, lines 8-25, column 15, lines 5-15, and column 19, lines 40-55, all of which discuss compensation routines and routines that allow a system to adjust and recoup from system failure to ensure reliable completion of the long running transaction. The system stores in a log the initiation, execution, and termination of each workflow action/step, which allows for easier recovery and resumption of subsequent workflow actions/steps upon these subsequent workflow actions/steps’ failure. Thus, the first workflow action is compensated based on the termination of the second, prior, workflow action (based on the log of its initiation, execution, and termination).

In response to argument 4), Examiner respectfully disagrees. Examiner first notes that both claims 51 and 52 are method claims, where the last three limitations state “if the action state is aborted”. Therefore, in instances in Hsu et al. where the workflow steps/action do not abort

for any reason (See column 3, lines 10-30), these last three limitations do not occur as the " if the action state is aborted" limitation is not satisfied.

In the instances where these limitations do occur however, Hsu et al. teaches both claims 51 and 52, which are substantially similar claims. First, Hsu et al. teaches determining the action state of the workflow action/step of the workflow. See column 3, lines 10-25, column 5, lines 23-30 and 60-67, column 8, lines 30-45, column 13, lines 25-40 and 50-60, wherein the stat of the workflow, the inputs, outputs, and log, are all considered. If the action state (status of the step) is aborted (stopped or timed out), determining the relationship of the workflow action (step) and the transaction (overall long running workflow) based on a transaction boundary (the limit being crossed and the transaction failing/timing out). See abstract, column 1, line 15-30, column 7, lines 5-25 and 50-60, column 14, lines 1-20. If the action state is aborted (as discussed just previously), and if the workflow action and transaction are related according to the transaction boundary (i.e. the workflow step is a step within the overall transaction/overall long running workflow where the workflow step has crossed a limit associated with the overall long running workflow based on the workflow having failed/having timed out), determining the transaction state of the transaction. See again column 2, line 64-column 3, line 7 and lines 25-30, column 4, lines 65-67, column 7, lines 8-25, column 15, lines 5-15, and column 19, lines 40-55, where a system will adjust and recoup from system failure to ensure reliable completion of the long running transaction. The system stores in a log the initiation, execution, and termination of each workflow action/step, which allows for easier recovery and resumption of subsequent workflow actions/steps upon these subsequent workflow actions/steps' failure.

If the action state is aborted and if the workflow action and the transaction are related according to the transaction boundary (discussed above), and if the transaction state is committed (i.e. the step was previously being executed and has timed out), performing an operation according to the compensation routine associated with the transaction. See again column 2, line 64-column 3, line 7 and lines 25-30, column 4, lines 65-67, column 7, lines 8-25, column 15, lines 5-15, and column 19, lines 40-55, all of which discuss compensation routines and routines that allow a system to adjust and recoup from system failure to ensure reliable completion of the long running transaction. Using the system log discussed previously, the action is compensated based on the log of its initiation, execution, and termination.

Thus, Hsu et al. dies teach and suggest each and every of the claim limitations for the reasons discussed above.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Beth Van Doren
Primary Examienr
AU 3623

Conferees:

Vincent Millin
Appeals Conference Specialist
TC 3600

Romain Jeanty
Primary Examiner
AU 3623